IDENTIFICATION OF WEB SITES THAT CONTAIN SESSION IDENTIFIERS

BACKGROUND OF THE INVENTION

A. Field of the Invention

[0001] The present invention relates generally to content retrieval on the world wide web, and more particularly, to automated web crawling.

B. Description of the Related Art

[0002] The World Wide Web ("web") contains a vast amount of information. Search engines assist users in locating desired portions of this information by cataloging web pages. Typically, in response to a user's request, the search engine returns references to documents relevant to the request.

[0003] Search engines may base their determination of the user's interest on search terms (called a search query) entered by the user. The goal of the search engine is to identify links to high quality relevant results based on the search query. Typically, the search engine accomplishes this by matching the terms in the search query to a corpus of pre-stored web documents. Web documents that contain the user's search terms are considered "hits" and are returned to the user.

[0004] The corpus of pre-stored web documents may be stored by the search engine as an index of terms found in the web pages. Documents that are to be added to the index may be automatically located by a program, sometimes

referred to as a "spider," that automatically traverses ("crawls") web documents based on the uniform resource locators (URLs) contained in the web documents. Thus, for example, a spider program may, starting at a given web page, download the page, index the page, and gather all the URLs present in the page. The spider program may then repeat this process for the web pages referred to by the URLs. In this way, the spider program "crawls" the world wide web based on its link structure.

[0005] Some web sites track users as they download different pages on the web site. User tracking is useful for identifying user behavior, such as identifying purchasing behavior by tracking the user through various web site page requests on a shopping orientated web site.

[0006] Two methods are commonly used to track user behavior: use of cookies to maintain information and embedding session identifiers in the uniform resource locators (URLs) in the web pages presented to the user. An embedded session identifier, in particular, may include a string of random characters embedded in the URLs returned to the user. Specifically, when the user requests a page of a web site with a URL that does not have a session identifier, a session identifier is created for this user and the user receives a version of the entry web page in which links on the page are annotated by the session identifier. When the user selects a link, the web server parses the session identifier from the URL, attaches the same session identifier to the local links on the next generated web page, and returns that web page to the user. The web

server continues to parse and attach the session identifiers as long as the user requests a page whose URL has a session identifier.

[0007] As an example of the use of session identifiers, consider the situation of a web spider crawling a first web site that contains multiple URLs that do not include session identifiers. The spider may decide to crawl these URLs, which may point to a second web site, one after the other. For each URL it requests, the spider may return a page whose URLs are annotated with a session identifier. Each requested page, however, may include a different session identifier. The spider would then extract these annotated URLs from the pages and if two URLs are identical except for the session identifier the spider would not recognize this since the URL strings are different. The spider would thus repeatedly crawl the same web pages, thus wasting the spider's time and bandwidth and filling the search engine's index with duplicate pages, thus wasting storage space.

[0008] Thus, there is a need in the art to effectively identify web sits that contain session identifiers in order to improve web crawling.

SUMMARY OF THE INVENTION

[0009] The present invention is directed to techniques for identifying web sites that use session identifiers.

[0010] A first aspect of the invention is directed to a method for crawling documents. The method includes receiving a uniform resource locator (URL) and determining whether the URL is associated with a web site that uses session

identifiers. The determination is made, at least in part, on a comparison of a portion of URLs that change between different copies of a web document downloaded from the web site.

[0011] A second aspect consistent with the invention is directed to a method for identifying web sites that use session identifiers. The method includes downloading at least two different copies of at least one web document from a web site, extracting URLs from the two different copies of the web document, comparing the extracted URLs of the two different copies of the web document, and determining whether the web site uses session identifiers based on the comparison.

[0012] Yet another aspect of the invention includes a device comprising a spider component configured to crawl web documents associated with at least one web site. Further, a session identifier component determines whether the web site uses session identifiers based on a calculation of a portion of URLs that change between different copies of at least one web document downloaded from the web site.

BRIEF DESCRIPTION OF THE DRAWINGS

[0013] The accompanying drawings, which are incorporated in and constitute a part of this specification, illustrate an embodiment of the invention and, together with the description, explain the invention. In the drawings,

[0014] Fig. 1 is an exemplary diagram of a network in which systems and methods consistent with the principles of the invention may be implemented;

[0015] Fig. 2 is an exemplary diagram of a client or server device according to an implementation consistent with the principles of the invention;

[0016] Fig. 3 is an exemplary functional block diagram illustrating an implementation of the server software shown in Fig. 1;

[0017] Fig. 4 is a flow chart illustrating the use of session identifiers by a web server;

[0018] Fig. 5 is a diagram illustrating a series of exemplary URLs that include session identifiers;

[0019] Fig. 6 is a flow chart illustrating operations consistent with aspects of the invention through which the session ID locator shown in Fig. 3 may classify web sites as sites that use session identifiers;

[0020] Fig. 7 is a diagram illustrating exemplary sets of links that correspond to a first and second set of links downloaded from a home page; and

[0021] Fig. 8 is a flowchart illustrating operations for crawling the web consistent with aspects of the invention.

DETAILED DESCRIPTION

[0022] The following detailed description of the invention refers to the accompanying drawings. The detailed description does not limit the invention.

[0023] As described herein, web sites that use session identifiers are determined. These web sites may be further analyzed to determine rules that describe the session identifier. A web crawler may use this information to enhance web crawling.

EXEMPLARY NETWORK OVERVIEW

[0024] Fig. 1 is an exemplary diagram of a network 100 in which systems and methods consistent with the principles of the invention may be implemented. Network 100 may include multiple clients 110 connected to one or more servers 120 via a network 140. Network 140 may include a local area network (LAN), a wide area network (WAN), a telephone network, such as the Public Switched Telephone Network (PSTN), an intranet, the Internet, or a combination of networks. Two clients 110 and a server 120 have been illustrated as connected to network 140 for simplicity. In practice, there may be more or fewer clients and servers. Also, in some instances, a client may perform the functions of a server and a server may perform the functions of a client.

[0025] Clients 110 may include client entities. An entity may be defined as a device, such as a wireless telephone, a personal computer, a personal digital assistant (PDA), a lap top, or another type of computation or communication device, a thread or process running on one of these devices, and/or an object executable by one of these device. Server 120 may include server entities that process, search, and/or maintain documents in a manner consistent with the principles of the invention. Clients 110 and server 120 may connect to network 140 via wired, wireless, or optical connections.

[0026] Clients 110 may include client software such as browser software 115. Browser software 115 may include a web browser such as the existing Microsoft Internet Explorer or Netscape Navigator browsers. For example, when network 140 is the Internet, clients 110 may navigate the web via browsers 115.

[0027] Server 120 may operate as a web server and include appropriate web server software 125. In one implementation, web server software 125 may function as a search engine, such as a query-based web page search engine. In general, in response to client requests, search engine 125 may return sets of documents to clients 110. The documents may be returned to clients 110 as a web page containing a list of links to web pages that are relevant to the search query. This list of links may be ranked and displayed in an order based on the search engine's determination of relevance to the search query. Although server 120 is illustrated as a single entity, in practice, server 120 may be implemented as a number of server devices.

[0028] A document, as the term is used herein, is to be broadly interpreted to include any machine-readable and machine-storable work product. A document may be an email, a file, a combination of files, one or more files with embedded links to other files, a news group posting, a web advertisement, etc. In the context of the Internet, a common document is a Web page. Web pages often include content and may include embedded information (such as meta information, hyperlinks, etc.) and/or embedded instructions (such as Javascript, etc.).

EXEMPLARY CLIENT/SERVER ARCHITECTURE

[0029] Fig. 2 is an exemplary diagram of a client 110 or server 120 according to an implementation consistent with the principles of the invention. Client/server 110/120 may include a bus 210, a processor 220, a main memory 230, a read

only memory (ROM) 240, a storage device 250, one or more input devices 260, one or more output devices 270, and a communication interface 280. Bus 210 may include one or more conductors that permit communication among the components of client/server 110/120.

[0030] Processor 220 may include any type of conventional processor or microprocessor that interprets and executes instructions. Main memory 230 may include a random access memory (RAM) or another type of dynamic storage device that stores information and instructions for execution by processor 220. ROM 240 may include a conventional ROM device or another type of static storage device that stores static information and instructions for use by processor 220. Storage device 250 may include a magnetic and/or optical recording medium and its corresponding drive.

[0031] Input device(s) 260 may include one or more conventional mechanisms that permit a user to input information to client/server 110/120, such as a keyboard, a mouse, a pen, voice recognition and/or biometric mechanisms, etc. Output device(s) 270 may include one or more conventional mechanisms that output information to the user, including a display, a printer, a speaker, etc. Communication interface 280 may include any transceiver-like mechanism that enables client 110/120 to communicate with other devices and/or systems. For example, communication interface 280 may include mechanisms for communicating with another device or system via a network, such as network 140.

[0032] The software instructions defining server software 125 and browser software 115 may be read into memory 230 from another computer-readable medium, such as data storage device 250, or from another device via communication interface 280. The software instructions contained in memory 230 causes processor 220 to perform processes that will be described later. Alternatively, hardwired circuitry may be used in place of or in combination with software instructions to implement processes consistent with the present invention. Thus, implementations consistent with the principles of the invention are not limited to any specific combination of hardware circuitry and software.

[0033] As mentioned, server software 125 may implement a search engine that, based on a user query, returns a list of links to documents that the server software 125 considers to be relevant to the search.

SERVER SOFTWARE 125

[0034] Fig. 3 is an exemplary functional block diagram illustrating an implementation of server software 125. Server software 125 may include a search component 305, a database component 310, a spider component 315, a session identifier (ID) locator 320, and a session ID rule generator 325. In general, search component 305 may receive user search queries from clients 110, search database 310 based on the search queries, and return a list of links (e.g., URLs) of relevant documents to the client 110. The list of links may also include information that generally attempts to describe the contents of the web documents associated with the links. The list of links may be ordered based on

ranking values, generated by search component 305, which rates the links based on relevance.

[0035] Database component 310 may store an index of the web documents that have been crawled by spider component 315. Database component 310 may be updated as new web documents are crawled and added to database component 310. Database component 310 may be accessed by search component 305 when responding to user search queries.

[0036] Spider component 315 may crawl documents available through network 140. Spider component 315 may include content filter 330, URL manager 335, and fetch bots 340. In general, fetch bots 340 may download content referenced by URLs. The URLs that are to be downloaded may be given to fetch bots 340 by URL manager 335. URL manager 335 may keep track of the URLs that have been downloaded and what URLs are to be downloaded. URL manager 335 may generate a "fingerprint" of each URL it receives by, for example, applying a hash function to the URL. The fingerprint can be used to quickly identify if a later-received URL is identical to one previously downloaded. Content filter 330 may receive content downloaded by fetch bots 340 and destined for database component 310. Content filter 330 may process the content downloaded by fetch bots 340 to, for example, extract URLs from the content for URL manager 335. Content filter 330 may forward the downloaded content to database component 310.

[0037] Session ID locator 320 may operate to identify web sites that use session identifiers. The identified web sites may be communicated to URL

manager 335, which may use this information when determining web documents to crawl. For instance, if the URL for a web document refers to a web host that uses session identifiers, URL manager 335 may first remove the session identifier from the URL before fingerprinting and/or storing the URL.

[0038] Session ID rule generator 325 may generate rules for URLs of web sites identified by session ID locator 320. These rules may describe how a particular web site inserts session identifiers into URLs that are embedded in web pages served by the web site and that refer to in-host resources (i.e., URLs that refer back to the same web site). URL manager 335 may use these rules to extract or insert session identifiers into URLs for a particular web site. The operation of session ID locator 320 and session ID rule generator 325 are described in more detail below.

[0039] Although search component 305, database component 310, spider component 315, session ID locator 320, and session ID rule generator 325 are illustrated in Fig. 3 as all being part of server software 125, one of ordinary skill in the art will recognize that these components could be implemented on separate computing devices or clusters of computing devices. Spider component 315 and search component 305, for example, may be implemented independently of one another. Additionally, session ID locator 320 and session ID rule generator 325 may also be implemented independently of spider component 315, database component 310, and/or search component 305.

[0040] Before describing the operation of the components in Fig. 3 in greater detail, it may be helpful to describe the way in which session identifiers are

commonly used. Fig. 4 is a flow chart illustrating the use of session identifiers by a web server. As mentioned, session identifiers may be used to track the user of a client 110 as the user interacts with a particular host web site.

[0041] A user may begin by browsing a web site using browser software 115 (act 401). The web site may assign a session identifier to the user (act 402). The session identifier may be, for example, a string of characters, such as a random string of sixteen or more hex-decimal characters. A web site that uses session identifiers to track user actions is typically a web site that includes multiple different possible web pages to which the user may navigate. A web shopping site, for instance, may contain hundreds or thousands of possible web pages to which the user may navigate. Typically, after accessing the main web page for the shopping site, the user will navigate to other web pages in the shopping site by selecting URLs on the main web page or from later served web pages. For each web page returned to the user, the web site may include the session identifier in the URLs that point to other web pages on the web site (act 403). When the user attempts to access one of these URLs, the web site uses the session identifier to identify the user with the user's previous site accesses.

[0042] Fig. 5 is a diagram illustrating a series of exemplary URLs that include session identifiers. A number of URLs 501-504 are illustrated in Fig. 5. In this example, URLs 501-504 are URLs embedded in a web page for "somecompany.com." Each URL includes a session identifier 510 ("12341234"). If the user selects URL 502, for instance, the user's browser program 115 may contact "somecompany.com" and request the web page "/12341234/page1.htm."

The "somecompany.com" web site may use session identifier 510, which the web server strips from URL 502, to identify the user's session. The web server also returns the actual web page, "page1.htm." The URLs within "page1.htm" that refer to other web pages at "somecompany.com" (i.e., the in-host URLs) may also contain session identifier 510.

[0043] Although session identifiers were described above as a specially inserted strings of characters, in general, any portion of a URL that doesn't reference content can be potentially used as a session identifier. Content, in this sense, does not necessarily mean that the web documents need to be exactly the same to be considered as having the same content. For example, web documents that are the same but for different color schemes, different advertisement links, or different navigation links may still be considered as having the same content.

[0044] Fig. 6 is a flow chart illustrating operations consistent with aspects of the invention through which session ID locator 320 may classify web sites as sites that use session identifiers.

[0045] Session ID locator 320 may begin with a list of web sites that are suspected to use session identifiers (act 601). In one implementation, the list may be manually generated and supplied to session ID locator 320. In alternate implementations, session ID locator 320, or another component such as URL manager 335, may compile the list automatically. URL manager 335, for example, may add entries to the list that correspond to web sites suspected to contain session identifiers that it is attempting to crawl. For example, a web site

for which URL manager 335 receives an exceptionally large number of "different" links may be considered to be a suspicious web site.

[0046] For web sites in the list of suspicious sites, session ID locator 320 may fetch (download) the home page of the corresponding web site two different times (act 602). The home page of a web site can normally be accessed by initiating a hyper-text transfer protocol (HTTP) request with the host name. For example, the home page of the web site corresponding to the host "google.com" may be downloaded with the HTTP request "http://www.google.com". In some implementations, other web documents at the web sites, instead of the home page, may be fetched instead.

[0047] If the web site under consideration is one that uses session identifiers, each of the two requests in act 602 should be returned with a different set of session identifiers. Thus, for example, with reference to Fig. 5, the first request to "somecompany.com" may return a web page with links 501-504, each including the session identifier "12341234." The second request to "somecompany.com" may return a web page with links similar to links 501-504, but including a different session identifier, such as "12345678."

[0048] Session ID locator 320 may extract the set of local (also referred to as in-host) links (URLs) from the web pages downloaded in act 602 (act 603). Local links are defined as links that refer back to the web host. The exemplary links 501-504 are all local links because they all refer to documents at "somecompany.com." If one of the links downloaded from the home page of "somecompany.com" referred to another web host, such as the exemplary link

"http://someothercompany.com/home.htm," this link may not be considered a local link by session ID locator 320.

[0049] Fig. 7 is a diagram illustrating exemplary sets of links that correspond to a first and second set of links downloaded from a home page. In this hypothetical example, the homepage is given as "http://somecompany.com/home.htm." The first set of links, set 701, includes four in-host links 702 and an external link 703. The second set of links, set 710, also includes four in-host links 712 and external link 713.

[0050] Session ID locator 320 may next compute the fraction of links that change by comparing the four in-host links 702 to the four in-host links 712 (act 604). In this example, the first three of the four links in sets 702 and 712 contain session identifiers that change between sets 702 and 712. The last link in sets 702 and 712 does not change between sets. Accordingly, three out of the four links (75%) change between links 702 and 712.

[0051] If the fraction determined in act 604 is above a predetermined threshold, session ID locator 320 may classify the corresponding web site as one that uses session identifiers (acts 605 and 606). The value to use as the threshold may be determined based on manual inspection of typical values calculated for a number of web sites.

[0052] Other techniques may be used to classify web sites as ones that use session identifiers. For example, if there is at least one link that changes between the two versions of the web page and the content underlying these two links are duplicates or near-duplicates of one another, the web site may be

considered to be one that uses session identifiers. Another technique may be to crawl URLs from a given web site until a certain number of pages have been crawled (e.g., 100). At this point, near-duplicate pages could be determined for the crawled pages. If the portion of near-duplicates is greater than a predetermined level, the web site could be considered to be one that uses session identifiers.

[0053] For sites that session ID locator 320 determines to use session identifiers, session ID rule generator 325 may analyze the links with session identifiers to determine a rule(s) that the site uses to insert the session identifiers (act 607). For the "somecompany.com" example shown in Fig. 7, for example, the rule may be determined as "insert the session identifier after the host name and delineate the session identifier with '/' characters." In some implementations, session ID rule generator 325 may be implemented manually by human operators. In other implementations, automated pattern classification techniques may be used to implement session ID rule generator 325.

[0054] Fig. 8 is a flowchart illustrating operations for crawling the web consistent with aspects of the invention. URL manager 335 may begin by identifying candidate URLs to crawl from previously crawled documents (act 801). The candidate URLs may include any URLs that were previously extracted from downloaded web documents. Alternatively, candidate URLs could be externally input to URL manager 335.

[0055] Using the techniques discussed above, session ID locator 320 may determine if a candidate URL is from a web site that uses session identifiers (act

802). If the URL is from a site that uses session identifiers, URL manager 335 may retrieve the session identifier rules generated by session ID rule generator 325 (act 803). URL manager 335 may use the rules to extract the session identifier from the candidate URL, thus generating a clean version of the URL (act 804).

[0056] URL manager 335 may determine if the candidate URL has been previously crawled by comparing the clean version of the candidate URL to clean versions of URLs that were previously crawled (act 805). The comparison may be based on a fingerprint of the URLs.

[0057] Candidate URLs that URL manager 335 determines should be crawled may be transmitted to fetch bots 340 for downloading (act 806).

CONCLUSION

[0058] As discussed above, web sites that use session identifiers may be automatically identified by comparing in-host links to multiple copies of documents from the web sites. Knowing that a particular web site uses session identifiers can enhance web crawling.

[0059] It will be apparent to one of ordinary skill in the art that aspects of the invention, as described above, may be implemented in many different forms of software, firmware, and hardware in the implementations illustrated in the figures. The actual software code or specialized control hardware used to implement aspects consistent with the present invention is not limiting of the present invention. Thus, the operation and behavior of the aspects were described

without reference to the specific software code--it being understood that a person of ordinary skill in the art would be able to design software and control hardware to implement the aspects based on the description herein.

[0060] The foregoing description of preferred embodiments of the present invention provides illustration and description, but is not intended to be exhaustive or to limit the invention to the precise form disclosed. Modifications and variations are possible in light of the above teachings or may be acquired from practice of the invention. For example, although the above discussion relating to determining if a web site uses session identifiers was described with reference to a single web site, a web site can be broadly defined to cover sites served by multiple web hosts. In this situation, the host name part of the URL for the web site may be different.

[0061] No element, act, or instruction used in the description of the present application should be construed as critical or essential to the invention unless explicitly described as such. Also, as used herein, the article "a" is intended to include one or more items. Where only one item is intended, the term "one" or similar language is used.